We claim the following:

1. A method for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer, said method including the following steps:

(a) receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer;

(b) determining whether said new note event is a note-on event;

(c) if said new note event is a note-on event, adding said particular note to a notes-on list;

(d) if said new note event is not a note-on event, deleting said particular note from said notes-on list;

(e) determining how many notes are on said notes-on list;

(f) selecting an assignment table corresponding to the predetermined number of channels and how many notes are on said notes-on list;

(g) assigning notes to said channels pursuant to said assignment table and said notes-on list; and

(h) sending to said musical synthesizer a set of commands corresponding to the assignment of notes to channels.

2. The method of Claim 1, wherein step (e) is as follows;

(e) determining how many notes are on said notes-on list and if there is not at least one note on said notes-on list, issuing a note-off command to said musical synthesizer for any note currently being played on any channel.

3. A method for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer, said method including the following steps:

(a) receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer;

(b) determining whether said new note event is a note-on event;

(c) if said new note event is a note-on event, adding said particular note to a notes-on list;

(d) if said new note event is not a note-on event, deleting said particular note from said notes-on list;

(e) determining how many notes are on said notes-on list;

(f) selecting an assignment table corresponding to the predetermined number of channels and how many notes are on said notes-on list;

(g) assigning notes to said channels pursuant to said assignment table and said notes-on list; and

(h) sending to a channel commands buffer a set of commands corresponding to the assignment of notes to channels.

4. A method for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said method including the following steps:

(a) receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer;

(b) determining whether said new note event is a note-on event;

(c) if said new note event is a note-on event, adding said particular note to a notes-on list;

(d) if said new note event is not a note-on event, deleting said particular note from said notes-on list;

(e) determining how many notes are on said notes-on list;

(f) selecting an assignment table corresponding to the predetermined number of channels and how many notes are on said notes-on list;

(g) assigning notes to said channels pursuant to said assignment table and said notes-on list;

(h) doing the following on a channel by channel basis,

(1) determining whether the note assigned to the channel pursuant to step (g) is the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event;

(2) if the note assigned to the channel pursuant to step (g) is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining whether there is a note assigned to the channel pursuant to step (g), and if there is no note assigned to the channel, sending a note-off command to said musical synthesizer with respect to said channel and the note currently being played by that channel; and

(3) if there is a note assigned to said channel pursuant to step (g), which note is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining whether said note assigned to said channel pursuant to step (g) is the same as any of the notes which were assigned to any of the other of said

channels just prior to the reception of said signal indicating said new note event, and doing the following:

(i) if said note assigned to said channel is the same as any of the notes which were assigned to any of the other of said channels just prior to the reception of said signal indicating said new note event, sending a note-on command to said musical synthesizer with respect to said channel and said note, which command is accompanied by a soft attack instruction; and

(ii) if said note assigned to said channel is not the same as any of the notes which were assigned to any of the other of said channels just prior to the reception of said signal indicating said new note event, sending a note-on command to said musical synthesizer with respect to said channel and said note, which command is accompanied by a hard attack instruction.

5. A method for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said method including the following steps:

(a) receiving a signal indicating a new note event, wherein a new note event is one of the following two events

relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes to be played by the musical synthesizer and (ii) the deletion of said particular note from the notes to be played by said musical synthesizer;

(b) determining whether said new note event is a note-on event;

(c) if said new note event is a note-on event, adding said particular note to a notes-on list;

(d) if said new note event is not a note-on event, deleting said particular note from said notes-on list;

(e) determining how many notes are on said notes-on list;

(f) selecting an assignment table corresponding to the predetermined number of channels and how many notes are on said notes-on list;

(g) assigning notes to said channels pursuant to said assignment table and said notes-on list;

(h) doing the following on a channel by channel basis,

(1) determining whether the note assigned to the channel pursuant to step (g) is the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event;

(2) if the note assigned to the channel pursuant to step (g) is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining

whether there is a note assigned to the channel pursuant to step (g), and if there is no note assigned to the channel, sending a note-off command to a channel commands buffer with respect to said channel and the note currently being played by that channel; and

(3) if there is a note assigned to said channel pursuant to step (g), which note is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining whether said note assigned to said channel pursuant to step (g) is the same as any of the notes which were assigned to any of the other of said channels just prior to the reception of said signal indicating said new note event, and doing the following:

(i) if said note assigned to said channel is the same as any of the notes which were assigned to any of the other of said channels just prior to the reception of said signal indicating said new note event, sending a note-on command to said a channel commands buffer with respect to said channel and said note, which command is accompanied by a soft attack instruction; and

(ii) if said note assigned to said channel is not the same as any of the notes which were assigned to any of the other of said channels just prior to the reception of said signal indicating said new note event, sending a note-on command to

said a channel commands buffer with respect to said channel and said note, which command is accompanied by a hard attack instruction.

6. A method for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said method including the following steps:

(a) receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer;

(b) determining whether said new note event is a note-on event;

(c) if said new note event is a note-on event, adding said particular note to a notes-on list;

(d) if said new note event is not a note-on event, deleting said particular note from said notes-on list;

(e) determining how many notes are on said notes-on list;

(f) selecting an assignment table corresponding to the predetermined number of channels and how many notes are on said notes-on list;

(g) assigning notes to said channels pursuant to said assignment table and said notes-on list;

(h) doing the following on a channel by channel basis,

(1) determining whether the note assigned to the channel pursuant to step (g) is the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event;

(2) if the note assigned to the channel pursuant to step (g) is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining whether there is a note assigned to the channel pursuant to step (g), and if there is no note assigned to the channel, sending a note-off command to said musical synthesizer with respect to said channel and the note currently being played by that channel; and

(3) if there is a note assigned to said channel pursuant to step (g), which note is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, doing the following:

(i) if said new note event is not a note-on event, sending a note-on command to said musical synthesizer with respect to said channel and said

note, which command is accompanied by a soft attack instruction; and

(ii) if said new note event is a note-on event, sending a note-on command to said musical synthesizer with respect to said channel and said note, which command is accompanied by a hard attack instruction.

7. A method for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said method including the following steps:

(a) receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes to be played by the musical synthesizer and (ii) the deletion of said particular note from the notes to be played by said musical synthesizer;

(b) determining whether said new note event is a note-on event;

(c) if said new note event is a note-on event, adding said particular note to a notes-on list;

(d) if said new note event is not a note-on event, deleting said particular note from said notes-on list;

(e) determining how many notes are on said notes-on list;

(f) selecting an assignment table corresponding to the predetermined number of channels and how many notes are on said notes-on list;

(g) assigning notes to said channels pursuant to said assignment table and said notes-on list;

(h) doing the following on a channel by channel basis,

(1) determining whether the note assigned to the channel pursuant to step (g) is the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event;

(2) if the note assigned to the channel pursuant to step (g) is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining whether there is a note assigned to the channel pursuant to step (g), and if there is no note assigned to the channel, sending a note-off command to a channel commands buffer with respect to said channel and the note currently being played by that channel; and

(3) if there is a note assigned to said channel pursuant to step (g), which note is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, doing the following:

(i) if said new note event is not a note-on event, sending a note-on command to said channel commands buffer with respect to said channel and said note, which command is accompanied by a soft attack instruction; and

(ii) if said new note event is a note-on event, sending a note-on command to said channel commands buffer with respect to said channel and said note, which command is accompanied by a hard attack instruction.

8. A method for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said method including the following steps:

(a) receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer;

(b) determining whether said new note event is a note-on event;

(c) if said new note event is a note-on event, adding said particular note to a notes-on list;

(d) if said new note event is not a note-on event, deleting said particular note from said notes-on list;

(e) determining how many notes are on said notes-on list;

(f) selecting an assignment table corresponding to the predetermined number of channels and how many notes are on said notes-on list;

(g) assigning notes to said channels pursuant to said assignment table and said notes-on list;

(h) storing an old note/channel list reflecting the assignment of notes to channels just prior to the reception of said signal indicating said new note event;

(i) doing the following on a channel by channel basis,

(1) determining whether the note assigned to the channel pursuant to step (g) is the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event;

(2) if the note assigned to the channel is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining whether there is a note assigned to the channel pursuant to step (g), and if there is no note assigned to the channel, storing an a note-off command in a channel commands buffer with

respect to said channel and the note currently being played by that channel; and

(3) if there is a note assigned to said channel pursuant to step (g), which note is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining whether said note assigned to said channel pursuant to step (g) is the same as any of the notes on the old note/channel list, and doing the following:

(i) if said note assigned to said channel is the same as any of the notes on the old note/channel list, storing a note-on command with respect to said channel and said note, which command is accompanied by a soft attack instruction, in said channel commands buffer; and

(ii) if said note assigned to said channel is not the same as any of the notes on the old/note channel list, storing a note-on command with respect to said channel and said note, which command is accompanied by a hard attack instruction, in said channel commands buffer; and

(j) sending to said musical synthesizer the commands stored in said channel commands buffer when step (i) is completed with respect to all of said predetermined number of channels.

9. The method of Claim 8 wherein said musical synthesizer is a sampler.

10. The method of Claim 9 including the additional step of storing a separate library for each of said channels.

11. A method for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said method including the following steps:

(a) receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer;

(b) determining whether said new note event is a note-on event;

(c) if said new note event is a note-on event, adding said particular note to a notes-on list;

(d) if said new note event is not a note-on event, deleting said particular note from said notes-on list;

(e) determining how many notes are on said notes-on list;

(f) selecting an assignment table corresponding to the predetermined number of channels and how many notes are on said notes-on list;

(g) assigning notes to said channels pursuant to said assignment table and said notes-on list;

(h) storing an old note/channel list reflecting the assignment of notes to channels just prior to the reception of said signal indicating said new note event;

(i) doing the following on a channel by channel basis,

(1) determining whether the note assigned to the channel pursuant to step (g) is the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event;

(2) if the note assigned to the channel is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, determining whether there is a note assigned to the channel pursuant to step (g), and if there is no note assigned to the channel, storing an a note-off command in a channel commands buffer with respect to said channel and the note currently being played by that channel; and

(3) if there is a note assigned to said channel pursuant to step (g), which note is not the same as the note which was assigned to said channel just prior to the reception of said signal indicating said new note event, doing the following:

(i) if said new note event is not a note-on event, storing a note-on command with respect to said channel and said note, which command is accompanied by a soft attack instruction, in said channel commands buffer; and

(ii) if said new note event is a note-on event, storing a note-on command with respect to said channel and said note, which command is accompanied by a hard attack instruction, in said channel commands buffer; and

(j) sending to said musical synthesizer the commands stored in said channel commands buffer when step (i) is completed with respect to all of said predetermined number of channels.

12. The method of Claim 11 wherein said musical synthesizer is a sampler.

13. The method of Claim 12 including the additional step of storing a separate library for each of said channels.

14. A system for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer after said system's receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer, wherein each channel of said musical

synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said system including,

(a) a CPU,

(b) a notes-on list memory location for storing a list of notes to be played (said list of notes to be played hereinafter referred to as a "notes-on list"),

(c) an assignment tables memory location for storing assignment tables,

(d) an old note/channel list memory location for storing a list of notes and corresponding channels to which said notes were assigned prior to said system's receiving said signal indicating a new note event,

(e) a new note/channel list memory location for storing a list of notes and corresponding channels to which said notes are assigned after said system's receiving said signal indicating a new note event,

(f) a channel comparison counter for counting channel comparisons done after said system's receiving said signal indicating a new note event, and

(g) a channel commands buffer,

wherein said CPU does the following:

(1) said CPU adds said particular note to the notes-on list if the new note event is a note-on event and removes said particular note from said note-on list if said new note event is not a note-on event;

(2) said CPU selects, depending on (i) how many notes are on the notes-on list and (ii) the predetermined number of channels, an assignment table from said assignments table memory location;

(3) said CPU assigns the notes listed on said notes-on list to said channels in accordance with the selected assignment table and creates a new note/channel list of the notes listed on the note-on list and the corresponding channels to which they have been assigned;

(4) said CPU stores said new note/channel list in said new note/channel list memory location;

(5) with said channel comparison counter set to one, said CPU begins comparing the new note/channel list to the old note channel list on a channel by channel basis;

(6) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is the same as the note assigned to that same channel on the old note/channel list, said CPU determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU causes the commands in the channel commands buffer to be sent to said musical synthesizer; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the

channel corresponding to the incremented value of the channel comparison counter;

(7) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if no note is assigned to said channel on the new note/channel list, said CPU sends a note-off command with respect to said channel and the note currently played on that channel to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU causes the commands in the channel commands buffer to be sent to said musical synthesizer; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter;

(8) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if the note assigned to said channel on the new note/channel list is not the same as any note on the old note/channel, said CPU sends

a note-on command with respect to said channel and said note assigned to it on the new note/channel list, along with a hard attack instruction, to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU causes the commands in the channel commands buffer to be sent to said musical synthesizer; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter; and

(9) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if the note assigned to said channel on the new note/channel list is the same as any note on the old note/channel, said CPU sends a note-on command with respect to said channel and said note assigned to it on the new note/channel list, along with a soft attack instruction, to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU causes the commands in the channel commands buffer to be sent to

said musical synthesizer; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter; and

(10) said CPU stores said new note/channel list in said old note/channel memory location when said commands in the channel commands buffer are caused to be sent to said musical synthesizer.

15. The system of Claim 14 wherein said musical synthesizer is a sampler.

16. The system of Claim 15 wherein said sampler includes a separate library for each of said channels.

17. A system for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer after said system's receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said system including,

(a) a CPU,

(b) a notes-on list memory location for storing a list of notes to be played (said list of notes to be played hereinafter referred to as a "notes-on list"),

(c) an assignment tables memory location for storing assignment tables,

(d) an old note/channel list memory location for storing a list of notes and corresponding channels to which said notes were assigned prior to said system's receiving said signal indicating a new note event,

(e) a new note/channel list memory location for storing a list of notes and corresponding channels to which said notes are assigned after said system's receiving said signal indicating a new note event,

(f) a channel comparison counter for counting channel comparisons done after said system's receiving said signal indicating a new note event, and

(g) a channel commands buffer,

wherein said CPU does the following:

(1) said CPU adds said particular note to the notes-on list if the new note event is a note-on event and removes said particular note from said note-on list if said new note event is not a note-on event;

(2) said CPU selects, depending on (i) how many notes are on the notes-on list and (ii) the predetermined number of channels, an assignment table from said assignments table memory location;

(3) said CPU assigns the notes listed on said notes-on list to said channels in accordance with the selected assignment table and creates a new note/channel list of the notes listed on the note-on list and the corresponding channels to which they have been assigned;

(4) said CPU stores said new note/channel list in said new note/channel list memory location;

(5) with said channel comparison counter set to one, said CPU begins comparing the new note/channel list to the old note channel list on a channel by channel basis;

(6) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is the same as the note assigned to that same channel on the old note/channel list, said CPU determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU ceases channel comparisons; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter;

(7) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not

the same as the note assigned to that same channel on the old note/channel list, and if no note is assigned to said channel on the new note/channel list, said CPU sends a note-off command with respect to said channel and the note currently played on that channel to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU ceases channel comparisons; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter;

(8) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if the note assigned to said channel on the new note/channel list is not the same as any note on the old note/channel, said CPU sends a note-on command with respect to said channel and said note assigned to it on the new note/channel list, along with a hard attack instruction, to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU ceases channel

comparisons; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter; and

(9) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if the note assigned to said channel on the new note/channel list is the same as any note on the old note/channel, said CPU sends a note-on command with respect to said channel and said note assigned to it on the new note/channel list, along with a soft attack instruction, to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU ceases channel comparisons; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter; and

(10) said CPU stores said new note/channel list in said old note/channel memory location when the value of

the channel comparison counter is equal to the predetermined number of channels.

18. A system for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer after said system's receiving a signal indicating a new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said system including,

(a) a CPU,

(b) a notes-on list memory location for storing a list of notes to be played (said list of notes to be played hereinafter referred to as a "notes-on list"),

(c) an assignment tables memory location for storing assignment tables,

(d) an old note/channel list memory location for storing a list of notes and corresponding channels to which said notes were assigned prior to said system's receiving said signal indicating a new note event,

(e) a new note/channel list memory location for storing a list of notes and corresponding channels to which said

notes are assigned after said system's receiving said signal indicating a new note event,

(f) a channel comparison counter for counting channel comparisons done after said system's receiving said signal indicating a new note event, and

(g) a channel commands buffer,

wherein said CPU does the following:

(1) said CPU adds said particular note to the notes-on list if the new note event is a note-on event and removes said particular note from said note-on list if said new note event is not a note-on event;

(2) said CPU selects, depending on (i) how many notes are on the notes-on list and (ii) the predetermined number of channels, an assignment table from said assignments table memory location;

(3) said CPU assigns the notes listed on said notes-on list to said channels in accordance with the selected assignment table and creates a new note/channel list of the notes listed on the note-on list and the corresponding channels to which they have been assigned;

(4) said CPU stores said new note/channel list in said new note/channel list memory location;

(5) with said channel comparison counter set to one, said CPU begins comparing the new note/channel list to the old note channel list on a channel by channel basis;

(6) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is the same as the note assigned to that same channel on the old note/channel list, said CPU determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU causes the commands in the channel commands buffer to be sent to said musical synthesizer; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter;

(7) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if no note is assigned to said channel on the new note/channel list, said CPU sends a note-off command with respect to said channel and the note currently played on that channel to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU causes the commands in the channel commands buffer to be sent to said musical synthesizer; (ii) it is not,

said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter;

(8) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if the new note event is a note-on event, said CPU sends a note-on command with respect to said channel and said note assigned to it on the new note/channel list, along with a hard attack instruction, to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU causes the commands in the channel commands buffer to be sent to said musical synthesizer; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter; and

(9) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not

the same as the note assigned to that same channel on the old note/channel list, and if the new note event is not a note-on event, said CPU sends a note-on command with respect to said channel and said note assigned to it on the new note/channel list, along with a soft attack instruction, to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU causes the commands in the channel commands buffer to be sent to said musical synthesizer; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter; and

(10) said CPU stores said new note/channel list in said old note/channel memory location when said commands in the channel commands buffer are caused to be sent to said musical synthesizer.

19. The system of Claim 18 wherein said musical synthesizer is a sampler.

20. The system of Claim 19 wherein said sampler includes a separate library for each of said channels.

21. A system for assigning notes to be played by a musical synthesizer to a predetermined number of channels of said musical synthesizer after said system's receiving a signal indicating a

new note event, wherein a new note event is one of the following two events relating to a particular note: (i) the addition of said particular note (referred to herein as a "note-on event") to the notes being played by the musical synthesizer and (ii) the deletion of said particular note from the notes being played by said musical synthesizer, wherein each channel of said musical synthesizer is configured such that once said channel is commanded to play a particular note said channel continues playing said note until said channel is commanded to not play said note, said system including,

(a) a CPU,

(b) a notes-on list memory location for storing a list of notes to be played (said list of notes to be played hereinafter referred to as a "notes-on list"),

(c) an assignment tables memory location for storing assignment tables,

(d) an old note/channel list memory location for storing a list of notes and corresponding channels to which said notes were assigned prior to said system's receiving said signal indicating a new note event,

(e) a new note/channel list memory location for storing a list of notes and corresponding channels to which said notes are assigned after said system's receiving said signal indicating a new note event,

(f) a channel comparison counter for counting channel comparisons done after said system's receiving said signal indicating a new note event, and

(g) a channel commands buffer,

wherein said CPU does the following:

(1) said CPU adds said particular note to the notes-on list if the new note event is a note-on event and removes said particular note from said note-on list if said new note event is not a note-on event;

(2) said CPU selects, depending on (i) how many notes are on the notes-on list and (ii) the predetermined number of channels, an assignment table from said assignments table memory location;

(3) said CPU assigns the notes listed on said notes-on list to said channels in accordance with the selected assignment table and creates a new note/channel list of the notes listed on the note-on list and the corresponding channels to which they have been assigned;

(4) said CPU stores said new note/channel list in said new note/channel list memory location;

(5) with said channel comparison counter set to one, said CPU begins comparing the new note/channel list to the old note channel list on a channel by channel basis;

(6) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is the same as the note assigned to that same channel on the old note/channel list, said CPU determines whether the value of the channel comparison counter is equal to the

predetermined number of channels, and if (i) it is, said CPU ceases channel comparisons; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter;

(7) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if no note is assigned to said channel on the new note/channel list, said CPU sends a note-off command with respect to said channel and the note currently played on that channel to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU ceases channel comparisons; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter;

(8) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not

the same as the note assigned to that same channel on the old note/channel list, and if the note assigned to said channel on the new note/channel list is not the same as any note on the old note/channel, said CPU sends a note-on command with respect to said channel and said note assigned to it on the new note/channel list, along with a hard attack instruction, to said channel commands buffer and determines whether the value of the channel comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU ceases channel comparisons; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter; and

(9) if the comparison shows that the note assigned to a channel corresponding to the value of the channel comparison counter on the new note/channel list is not the same as the note assigned to that same channel on the old note/channel list, and if the note assigned to said channel on the new note/channel list is the same as any note on the old note/channel, said CPU sends a note-on command with respect to said channel and said note assigned to it on the new note/channel list, along with a soft attack instruction, to said channel commands buffer and determines whether the value of the channel

comparison counter is equal to the predetermined number of channels, and if (i) it is, said CPU ceases channel comparisons; (ii) it is not, said CPU increments the note channel counter by one and compares note/channel assignment on the new note/channel list with that on the old note/channel list for the channel corresponding to the incremented value of the channel comparison counter; and

(10) said CPU stores said new note/channel list in said old note/channel memory location when the value of the channel comparison counter is equal to the predetermined number of channels.